

Tracker Assisted Peer Scheduling Strategy in Multi-channel P2P VoD Streaming

T. Suresh, PhD
Assistant Professor

Department of Computer Science and Engineering
Annamalai University

K.T. Meena Abarna, PhD
Assistant Professor

Department of Computer Science and Engineering
Annamalai University

ABSTRACT

Peer-to-Peer Video-on-Demand (VoD) is a capable solution which offers thousands of videos to millions of users with complete interactive watching. Most of the commercial P2P streaming deployments PPLive, PPStream, UUSee have introduced a multi-channel P2P VoD system that allows user to view more than one channel at a time. Recent research studies have proposed a cross channel resource sharing algorithms to utilize the individual peer resources effectively, including bandwidth and cache capacity by enabling cross-channel cooperation among multiple channels. However, current multiple channel P2P VoD system deliver a video at a low streaming rate due to the channel resource imbalance and channel churn. In order to improve the streaming capacity, this paper proposes different effective helpers based resource balancing scheme that actively identifies the supply-and-demand imbalance in multiple channels. Furthermore, peers in a surplus channel serve its unused bandwidth resources to peers in a deficit channel that minimizes the server bandwidth consumption. This approach proposes a tracker assisted peer scheduling policy that effectively schedules the different chunks within each video in the process of fetching and serving chunks without impairing the streaming quality. Experimental evaluation shows that the proposed tracker assisted scheduling strategy achieves high streaming capacity under reduced server workload and improves streaming quality when compared to existing algorithms.

General Terms

Peer to Peer Networks, Video-on-demand, Multi-channel system.

Keywords

Cross channel cooperation, Tracker assisted peer scheduling policy.

1. INTRODUCTION

Video-on-demand (VoD) services have become extremely popular on the Internet today. VoD services have been attracting millions of users due to the sharing and watching of video at any time [1]. Traditional client/server VoD systems not able to provide the streaming quality in the pool of a large number of concurrent users and incurs high server workload. It is appealing to incorporate peer-to-peer (P2P) technology into VoD services that reduce the server workload by effectively utilizing the uplink bandwidth of the peers. Peer assisted video on-demand (PA-VoD) systems have demonstrated a considerable potential for exploiting the enormous amount of peer contributed resources such as peer upload bandwidth as well as cached content to serve other peers those who watching any other video in the system. This unique characteristic not only minimizes the server bandwidth consumption, it also reduces the upload bandwidth traffic in a large scale P2P VoD system [2]. This virtual quality of peer

contribution makes P2P VoD system capable of providing high streaming capacity with less maintenance cost when compared to the traditional client-server architecture.

Recently, commercial P2P streaming systems such as PPLive [3], PPStream [4], UUSee [5] have started to offer multiple video channels, where peers are able to watch more than one channel at a time according to their interest. Depending on the resource sharing, multi-channel P2P video streaming system can be classified into two types such as independent channel and correlated-channel P2P VoD systems. Typically, correlated channel systems effectively utilize the resources rather than independent channel systems. The streaming capacity of the multi-channel P2P streaming systems relies on the effective performance of the three techniques such as chunk scheduling, bandwidth allocation and caching mechanisms. Due to the heterogeneous nature of multi-channel P2P streaming, bandwidth availability across different channels is not uniform i.e., some channels have surplus bandwidth while other channels have deficit bandwidth. Therefore, cross channel resource allocation algorithm allocates the bandwidth among the multiple channels so that the peers in surplus channel shares its unused resources to the deficit channel which needs a bandwidth to satisfy playback rate. Next, Chunk scheduling mechanism is another equally important notion in P2P VoD system to improve the streaming capacity. It schedules the divided chunks for effectively delivery of the video streams to the peers within a playback deadlines. Last, caching mechanism predominantly caches the videos in order to serve future request with the objective of reducing the server workload and inter domain traffic in a large scale P2P VoD system.

2. RELATED WORKS

The main aim of this work is to improve the streaming capacity of a multi-channel P2P video system by crushing its two intrinsic characteristics: channel churn and channel resource imbalance. In recent years, lot of research activities carried out in P2P video streaming that fall into two categories: tree-based and mesh-based overlays. In the aspect of stability, mesh based P2P VoD achieves effective streaming performance because tree based overlays are not durable under dynamic churn. CoolStreaming/ (DONet) [6] constructs data driven overlay network on top of the Internet in order to utilize the peer upload bandwidth among the peers within the same overlay. This intra-overlay optimization scheme degrades the streaming performance due to its low resource utilization, high startup delay and inefficient resource assignment in global P2P networks. Therefore, Liao et al. [7] proposes effective intra-overlay cooperation scheme called AnySee that balance the resources among channels and also optimize streaming paths among the peers in different overlays. Yifeng *et al.* published three research works [8] [9] [10] that proposed chunk scheduling and optimal algorithms

to improve the streaming capacity in multi-channel P2P streaming systems. In [8], Yifeng et al. developed a proximal decomposition algorithm to optimize the streaming capacity problem in a distributed manner under peer bandwidth constraints. In [9], Yifeng et al. propose a fully distributed algorithm to maximize the aggregate throughput among all the peers in a scalable P2P VoD system. This approach achieves a higher throughput than proportional allocation scheme or the equal allocation scheme. In [10], helpers are introduced for the effective utilization of helpers upload bandwidth in order to serve a segment with a greedy algorithm. This work proposes helper assignment and rate allocation algorithms to improve the streaming capacity at the same time reduce the server workload.

While concerning the intra-channel and cross-channel resource allocation in multi-channel systems, Wu et al proposes two closely related papers to improve the server bandwidth utilization in multi-channel P2P streaming systems. In [11], Wu et al. proposed Ration, an online server capacity provisioning algorithm to adjust the supply of server bandwidth proactively among the different channels according to the forecasted demand, taking into account the number of peers, the streaming quality, and the priorities of channels. In [12], Wu et al. proposed a radically different cross-channel P2P streaming framework called View-Upload Decoupling. It effectively allocates the peer upload bandwidth allocation among the multiple channels by strictly decoupling the uploading content from the movie that a peer is viewing. This approach formulates the effective optimization algorithm to minimize the main two problems such as dynamic channel churn and channel resource imbalance among multiple channels. Therefore, this approach enables effective cross channel resource sharing and also achieves stability in multi-channel P2P streaming systems.

In order to analyze the server bandwidth consumption among the peers with either homogeneous or heterogeneous upload capacity, three models [13] [14] [15] are built in multi-channel VoD systems. Kumar et al. [13] develop a simple stochastic fluid model that seeks to expose the performance limitations of channel based P2P streaming systems. This model provides closed-form expressions to examine the fundamental characteristics of a P2P streaming system, including the dynamic churn, heterogeneous upload capacity, limited infrastructure capacity and peer buffering and playback delay. However, this study assumed that channels are of homogeneous playback rates. Moreover in [14], the same authors further extended a tractable infinite server queuing network model to capture heterogeneous peer channel switching patterns. This paper analytically studies the performance of multi-channel systems by considering peer channel switching, peer churn, bandwidth heterogeneity and zipf-like channel popularity. In [15], Linear programming model differs from the abovementioned two models because it is absolutely independent of implementations. This approach provides a closed form solutions due to its linear property that paves a way to explore the design space of multi-channel systems numerically. In [16], the linear programming model used to analyze the three bandwidth allocation schemes such as Naive Bandwidth allocation Approach (NBA), Passive Channel-aware bandwidth allocation Approach (PCA) and Active Channel-aware bandwidth allocation Approach

(ACA). They are analyzed in terms of streaming quality, channel structure, implementation complexity, peer population distributions and upload bandwidth distributions using bandwidth satisfaction ratio. Compared to NBA and PCA designs, ACA design provides effective performance since it allows a peer to subscribe to unwatched channels. However, this approach does not compare the system performance under peer dynamics (channel switching/peer leaving) in the transition state. Still, none of these proposals has taken collaboration of normal peers in multiple VoD channels into consideration.

3. TRACKER ASSISTED PEER SCHEDULING STRATEGY IN MULTI-CHANNEL P2P VoD SYSTEM

In this paper, we propose a typical multi-channel P2P VoD streaming system that providing multiple video channels to the peers with different upload capacities. The proposed approach improves the streaming capacity at the same time reduces the server workload by effectively sharing the bandwidth and cache capacity among the multiple channels. For cross channel bandwidth sharing, this work proposes effective helpers based resource sharing where peers in surplus channel shares its resources with the deficit channel to overwhelm the resource imbalance problem in heterogeneous multi-channel P2P VoD system. In order to utilize the cache space in a distributed manner effectively, it proposes a tracker assisted strategy that dynamically tunes the number of replicas according to the dynamic popularity. Fig 1 presents a high level architecture overview of proposed approach that comprises a streaming server, tracker and a set of client peers in the different channels. This section describes the function of the principal components in detail.

i) Streaming Server: Streaming server is the source of content that consists of persistent copy of a set of videos. It is a baseline of content availability provides the video for requesting peers. Video file is segmented into consecutive stream of chunks to perform VCR operations such as pause, forward and rewind.

ii) Client Peers: It consists of two components such as local buffer and the media player that enables to play the video. A peer fetches chunks from the streaming server or any other peers in different channels and keeps them in a linked list associated with the local buffer. The local buffer pulls fetched chunks of the video into the media player that decodes and plays a video in the end user side [17].

iii) Tracker: The tracker is a well-known rendezvous point to record the log information about the membership list of all situated peers in different channels. After fetching video from the server or any other peers in the system, meanwhile it must inform to the tracker about the cached data in its local disk to share resource with the requesting peers in the future. This information enables us to exchange the buffer map among the peers that recommend for video downloading based on its required chunks. In the future, the videos may be replicated across different peers that decrease both access latency and network traffic.

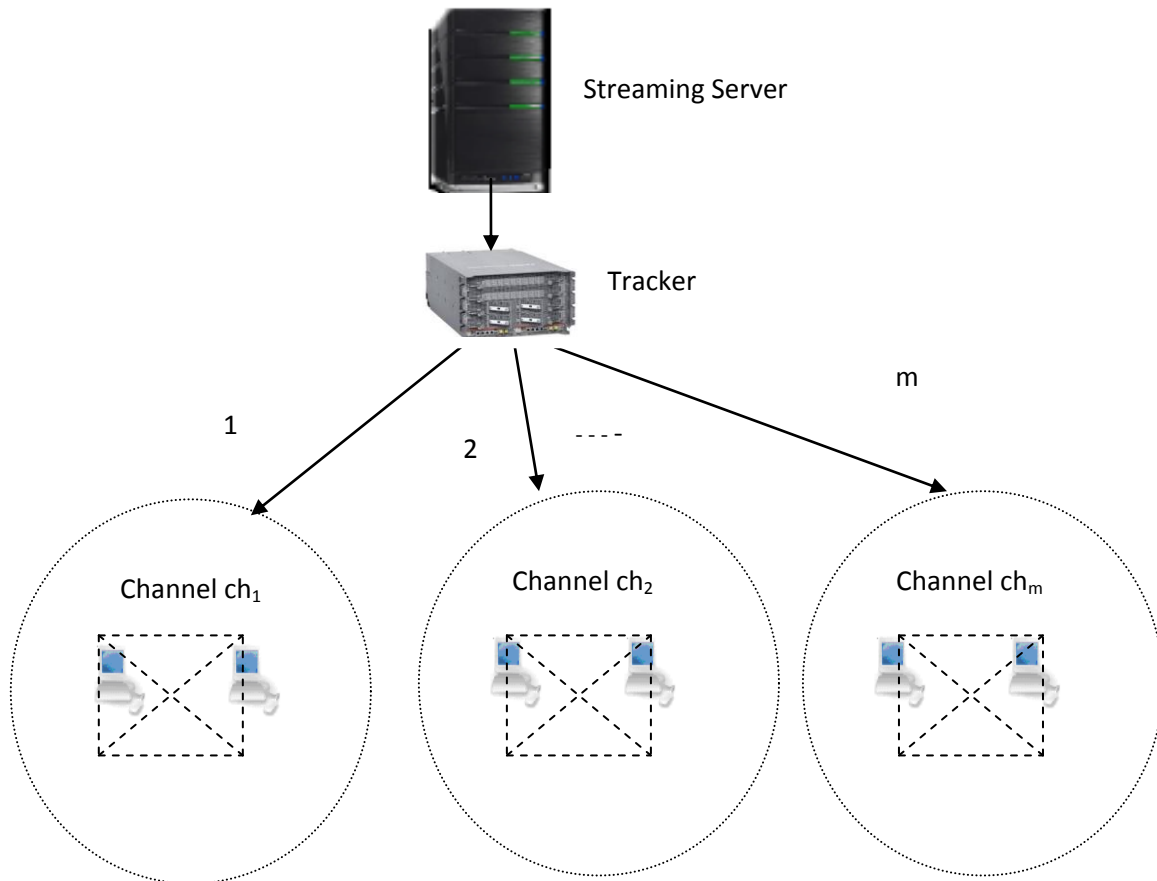


Fig 1: Tracker assisted Multi-channel P2P VoD streaming

3.1 Tracker Assisted Peer Scheduling Policy

This section depicts the peer scheduling policy to define the process of downloading a requested video (v_k) by peer (p_{req}) in both the downloader's and uploader's point of view. For downloader's view point, this policy aids to find which peers can provide the requested video with required streaming rate. For uploader's view point, it decides to upload the video for appropriate peers when receiving multiple requests. This paper proposes the tracker assisted peer scheduling policy that processes the query request generated by the peer through clicking the required video in the website or VSP. First, a requested peer looks for required video among the active peers who are viewing the data. If the playback rate of the video cannot be fulfilled by the active peer, then requested peer forwards query into the tracker. It returns a list of peers those who currently cache the requested videos and also redirects a query to the set of replication peers those who cache the requested video. Afterwards, the requested peer downloads the video from the replication peers in a P2P manner. Synchronized peer can support the VCR operations while playing the video in user side. However, it is difficult in the case of replication peer because it may caches the part of the video rather than whole. It results in frequent rescheduling when requested part of the video is not available in it. Therefore, tracker maintains the buffer map to support video downloading when required part of the video is not cached in replication peer. If the playback rate of video cannot be fulfilled by both synchronized and replication peers, the requesting peer directly streams the video from the streaming server in its required playback rate. The proposed approach downloads the video with less intra-domain traffic and totally

reduces the server workload due to the deployment of tracker for P2P assisted video streaming [18].

3.2 Cross-Channel Resource Sharing

In multi-channel P2P-VoD system, video channels are categorized based on available upload bandwidth resources into two types such as surplus channel and deficit channel. In surplus channel, the average upload capacity (U_c) of all peers is greater than playback rate P_r that is $U_c > P_r$. On the contrary, in deficit channel the average upload capacity U_c of all peers is less than the playback rate $U_c < P_r$. Therefore, peer in a surplus channel ch_1 shares the remaining upload capacity (U_{rem}) to the other peers in deficit channel to perform streaming. This aspect of resource sharing among the multiple channels improve the streaming capacity at the same time reduces the server workload. This work mainly aims to explore how a peer will dynamically and actively decide when it has sufficient bandwidth to serve for the peers in the other channel. Moreover, it also provides a mechanism to identify the peers which is in the need to get upload bandwidth from the other peers due to its deficit bandwidth.

3.2.1 Helpers Based Resource Balancing Scheme

This approach proposes cross channel helpers based resource balancing scheme to utilize the resources effectively among the multiple channels in a P2P VoD systems. Helper peer is the peer who will provide its unused remaining resource to peers of the other channels for effective resource utilization. First, tracker determines the set of peers with high upload bandwidth resources greater than the threshold. Then, it sorts the set of high resources peers according to its remaining upload resources. Among the sorted list of peers, a peer with

high bandwidth resources is selected as a helper for resource sharing [19]. Using the above procedure, peer (i) in surplus channel (ch_1) having remaining upload bandwidth ($U_{rem}(ch_1)$) selects as a cross channel helper ($H(c_{h1})$) to serve its upload resources to the peers in deficit channel (ch_2). Second, tracker determines the set of peers $\{P_1(c_{h2}), P_2(c_{h2}), \dots, P_i(c_{h2})\}$ who has insufficient bandwidth in deficit channel (ch_2) to satisfy its playback rate for viewing the segment j. It computes the channel resource index ($C_{Ch_2}(t)$) of the deficit channel as follows

$$C_{Ch_2}(t) = r_{ch_2} | H_{ch_2}(t) | - U_{ch_2}(t) \text{ ----- (1)}$$

Here, r_{ch_2} is the bit rate of channel (ch_2) and $H_{ch_2}(t)$ is the set of watching peers in the channel. And then, $U_{ch_2}(t)$ is the amount of bandwidth needed by the set of deficit peers in channel (ch_2) to perform streaming. Therefore, the peers who have less resource index to download the segment (i) at a required playback rate selected as a deficit peers. These peers require upload capacity from any of the peers in surplus channel to maintain the effective streaming quality. In this helper based resource balancing scheme, selected cross channel helper utilizes the remaining bandwidth in channel (ch_1) to download the segment (j) from the streaming server or any of the active peers watching the segment j at a streaming rate of $R_j(\text{download})$. After downloading the segment (j), cross channel helper upload into the set of peers which are waiting to view the video (j) in channel (ch_2) at a rate of $R_j(\text{upload})$. The bandwidth gain of the helper peers computed as follows

$$BG = R_j(\text{upload}) / R_j(\text{download}) \text{ ----- (2)}$$

In order to utilize the served bandwidth effectively, the streaming rate of downloading a segment (j) must be greater than the streaming rate of uploading a segment (j). Therefore, the bandwidth gain is always greater than 1. Therefore, the cross channel helper bandwidth contributes less bandwidth than it consumes from the streaming server [20]. The proposed resource sharing approach improves the streaming rate and also optimizes the server upload allocation for each channel by reducing the server load.

4. PERFORMANCE EVALUATION

This section carry out an extensive simulation to validate and evaluate the performance of proposed Tracker assisted peer Scheduling strategy with helpers based resource channel allocation approach. Experimental set up comprises a nearly 5,000 peers and streaming server provide nearly 200 videos with an upload capacity of 70Mbps. Each video plays at a playback rate (Pr) of = 500Kbps and its popularity follows Zipf distribution. The length of the video is 60 minutes, which is evenly divided into 60 segments. The peer's upload and download capacities uniformly distributed between in the range of 1.5 Mbps and 3.0 Mbps. Each peer maintains a buffer with a capacity of 5 segments. The playback time of each peer is distributed randomly between 0 and 60 minutes. Furthermore, experimental results show the effectiveness of the proposed approach in terms of streaming capacity and reduction of server load by comparing with the existing proportional approach.

4.1 Average Streaming Capacity

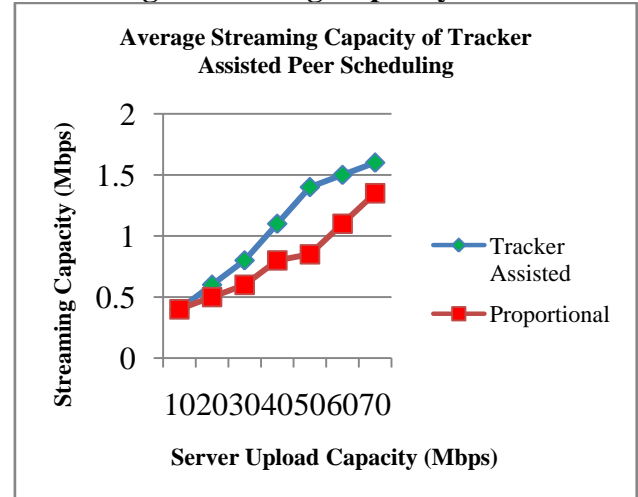


Fig 2: Average Streaming Capacity in Tracker assisted Peer Scheduling

Fig 2 shows that the proposed tracker assisted peer scheduling approach achieves high streaming capacity than the existing proportional approach in accordance to the increasing server upload capacity. In proposed approach, the streaming capacity increases gradually by optimally allocating the server upload capacity among the multiple channels using helpers based resource sharing scheme. Whereas in the existing scheme, peer bandwidth resources shares within the same channel, not among the multiple channels. Therefore, expected streaming quality cannot be acquired in proportional approach.

4.2 Reduction of Server Load

4.2.1 Effect of Cache Size

Fig 3 shows the reduction of server load in a proposed Tracker assisted peer scheduling approach. According to the increasing cache size, active peers itself can able to satisfy the request from any other peers in the system. Therefore, requested peer does not depend upon the server for streaming. Thus, the server workload can be predominantly reduced according to the increasing cache size. The following graph clearly examines that when the cache size is small; both the approaches come under the same server workload. In Tracker assisted, peers replicate videos in a smart and cooperative way to avoid repeatedly caching of videos that are likely to be cached by the other peers in the system. More importantly, when the cache is large enough, proposed Tracker assisted peer scheduling performs better than the existing proportional approach.

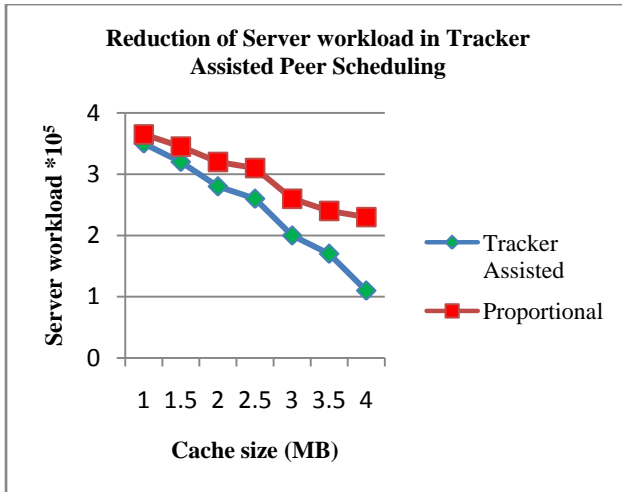


Fig 3: Reduction of server workload in Tracker assisted Peer Scheduling

4.2.2 Effect of the Number of Peers

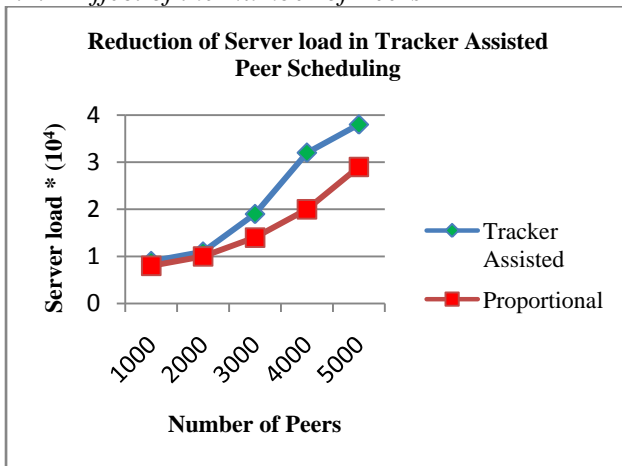


Fig 4: Reduction of server load in Tracker Assisted Peer Scheduling

Fig 4 shows that the Tracker assisted peer scheduling approach acquires less server workload than the already existing proportional approach even under a large number of peers. This is because; proposed approach signifies the effective replication of the number of replicas in a Tracker assisted manner that makes a better use of the peers' cache space. Therefore, effective maintaining of the videos in a cache space improves the streaming quality and also reduces the server workload.

5. CONCLUSION

This paper mainly considers the newly emerging multi-channel P2P video-on-demand streaming service and the problem of how to use a tracker collaboratively. It proposed a cross channel helpers based resource balancing scheme that effectively utilizes the resources to optimize the server upload allocation for each channel. The proposed algorithms are effective even under a dynamic churn and videos with different viewing popularity. Experimental evaluation illustrates that proposed approach achieves high streaming capacity and less server workload than existing proportional approaches. In future, the work lies on the deployment of peer scheduling policy to spread the video content among the large no of Multi-channel mobile users.

6. REFERENCES

- [1] C. Huang, J. Li, and K. W. Ross, "Can internet video-on-demand be profitable?", ACM SIGCOMM Computer Communication Review, Vol 37, No 4, pp 133-144, 2007.
- [2] Yan Huang, Tom Z. J. Fu, Dah-Ming Chiu, John C. S. Lui and Cheng Huang, "Challenges, Design and Analysis of a Large-scale P2P-VoD System", Proceedings of the ACM SIGCOMM 2008 conference on Data communication, pp 375-388, 2008.
- [3] PPLive <http://www.pplive.com/en/index.html>, accessed, 2013.
- [4] PPStream <http://www.ppstream.com/>, accessed, 2013.
- [5] UUsee <http://www.uusee.com>, accessed, 2013.
- [6] Xinyan Zhang, Jiangchuan Liu, Bo Liz, and Tak-Shing Peter Yum, "CoolStreaming/DONet: A data-driven overlay network for peer-to-peer live media streaming", Proceedings 24th Annual Joint Conference of the IEEE Computer and Communications Societies, Vol 3, pp 2102 - 2111, 2005.
- [7] Xiaofei Liao, Hai Jin, Yunhao Liu, Lionel M. Ni, and Dafu Deng, "AnySee: Peer-to-peer live streaming", Proceedings of 25th IEEE International Conference on Computer Communications, pp 1-10, 2006.
- [8] Yifeng He, Ivan Lee, Ling Guan, "Streaming capacity in P2P VoD systems", Proceedings of the IEEE International Symposium on Circuits and Systems, pp. 742-745, 2009.
- [9] Yifeng He, Ivan Lee, Ling Guan, "Distributed throughput maximization in P2P VoD applications," IEEE Transactions on Multimedia, Vol 11, No 3, pp. 509-522, 2009.
- [10] Yifeng He, Ivan Lee, Ling Guan, "Improving the streaming capacity in p2p VoD systems with helpers", Proceedings of the IEEE International Conference on Multimedia and Expo, pp 790-793, 2009.
- [11] C. Wu, B. Li, and S. Zhao, "Multi-channel live p2p streaming: Refocusing on servers", Proceedings of the IEEE international conference on Computer Communications, 2008.
- [12] D. Wu, C. Liang, Y. Liu, and K. Ross, "View-upload decoupling: a redesign of multi-channel P2P video systems," Proceedings of the IEEE 28th Conference on Computer Communications, pp. 2726-2730, April 2009.
- [13] R. Kumar, Y. Liu, and K. Ross, "Stochastic fluid theory for P2P streaming systems," Proceedings of the 26th IEEE International conference on Computer Communications, pp. 919-927, 2007.
- [14] D. Wu, Y. Liu, and K. W. Ross, "Queuing network models for multichannel P2P live streaming systems," Proceedings of the IEEE International conference on Computer Communications, pp. 73-81, 2009.
- [15] M. Wang, L. Xu, and B. Ramamurthy, "Linear programming models for multi-channel P2P streaming systems," Proceedings of the IEEE INFOCOM, pp. 1-5, 2010.
- [16] Miao Wang, LisongXu, and Byrav Ramamurthy, "Exploring the Design Space of Multichannel Peer-to-

- Peer Live Video Streaming Systems”, IEEE/ACM Transactions on Networking, Vol 21, No 1, pp 162-175, 2013.
- [17] Duc A. Tran, Kien A. Hua, Tai T. Do, “A Peer-to-Peer Architecture for Media Streaming”, IEEE Journal on Selected Areas in Communications, Vol 22, No 1, pp 121-123, 2004.
- [18] Y. Zhou, T.Z.J. Fu, et al., “A unifying model and analysis of P2P VoD replication and scheduling”, Proceedings of the IEEE INFOCOM, pp 1530 – 1538, 2012.
- [19] Z. Wang, C. Wu, L. Sun, and S. Yang, “Strategies of collaboration in multi-channel P2P VoD streaming”, IEEE Global Telecommunications Conference, pp 1-5, 2010.
- [20] Yifeng He and Ling Guan, “Improving Streaming Capacity in Multi-Channel P2P VoD Systems via Intra-Channel and Cross-Channel Resource Allocation”, International Journal of Digital Multimedia Broadcasting, Vol 2012, 2012.
- [21] W. Wu and J. Lui, “Exploring the optimal replication strategy in P2PVoD systems: characterization and evaluation”, In Proceedings of the 30th IEEE International Conference on Computer Communications, pp. 1206–1214, 2011.